A variational approach to empirical mode estimation

Tâm Le Minh, Julyan Arbel, Florence Forbes, Florence Forbes

A variational approach to empirical mode estimation

Tâm Le Minh[1], Julyan Arbel[1], Florence Forbes[1]

[1] *Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, 38000 Grenoble, France*

Abstract. Mode estimation, which locates the values at which a probability distribution is maximal, is a critical task in statistical analysis, with applications spanning inverse problems, clustering, and image analysis. While traditional methods focus on single-mode detection, identifying multiple modes is often essential for capturing the data underlying structure. We introduce empirical Natural Variational Annealing (eNVA) that can achieve this task for distributions possibly only available through samples. eNVA is an extension of the recent Natural Variational Annealing (NVA) optimization framework. Using the flexibility of a variational formulation, eNVA accommodates variational distributions within the exponential family and generalizes the Gaussian mean-shift algorithm. Furthermore, the multiple mode estimation challenge can be tackled efficiently with our eNVA-GM variant, using Gaussian mixtures (GM) as variational distributions and allowing adaptive search strategies, such as temperature annealing, to balance exploration and exploitation principles. Insights demonstrate the robustness and versatility of eNVA in addressing complex multimodal problems.

1 Introduction

Consider a smooth and bounded distribution p supported on the entire space $\mathbb{R}^d$, with a finite number of maxima:

$$\{\boldsymbol{x}_1^*, \ldots, \boldsymbol{x}_I^*\} = \arg\max_x p(\boldsymbol{x}).$$

The multimodal estimation problem aims to locate the multiple modes of p. Estimating the mode of a distribution is a fundamental problem in statistical analysis, with many applications in inverse problems, clustering (Chacón, 2015), and image segmentation (Carreira-Perpiñan, 2006). In many of these applications, identifying multiple modes, rather than just one, provides significant advantages. For instance, in Bayesian inverse problems, the diversity of modes of a posterior distribution can help improve the decision-making process. In clustering and image segmentation, identifying multiple modes allows for the detection of distinct groups or regions within the data, enabling better partitioning and representation of complex structures. In our knowledge, despite the long history of mode-estimation, most approaches are designed to locate a single mode of a distribution.

When a closed-form expression of p is available, modes can be determined by traditional optimization, locating the maxima of p seen as a function. In this work, we address *empirical* multiple mode estimation, considering the case where p is not explicitly available, but only through some of its samples.

Early methods, that have been designed to estimate modes from samples, consist in identifying intervals or regions where samples concentrate (Chernoff, 1964; Dalenius, 1965; Venter, 1967; Sager, 1978). These methods offer computational simplicity and robustness as they avoid any commitment to an intermediate, possibly imperfect, step of density estimation. However, their performance can depend on the choice of the geometric construct used to define the regions of high data concentration, and their extension to multiple modes or local modes presents additional challenges. By construction, they also lack the additional information about the distribution provided by density-based methods.

Alternatively, other methods estimate the mode by first approximating the density function and subsequently identifying the point where the estimated density achieves its maximum, going back to a standard optimization task. Examples of these approaches include kernel density estimation (KDE, Parzen, 1962), nearest-neighbor density estimation (Loftsgaarden and Quesenberry, 1965), orthogonal series expansions (Kronmal and Tarter, 1968), log-concave density estimation (Samworth, 2018), and Bernstein polynomials (Liu and Ghosh, 2020). These methods are particularly valuable because they provide not only the mode but also a comprehensive representation of the distribution, including its potential multimodal structure. For example, they can explore multiple density modes across varying levels of smoothing (Minnotte and Scott, 1993; Chaudhuri and Marron, 1999). However, this broader focus, suitable for multimodal estimation, comes at the expense of increased computational complexity, particularly in high-dimensional settings.

A third class of methods are mean-shift algorithms (Fukunaga and Hostetler, 1975; Comaniciu and Meer, 2002; Carreira-Perpiñan, 2007; Genovese et al., 2016; Arias-Castro et al., 2016). They perform gradient ascent on the KDE to locate modes. While rooted in KDE theory, these algorithms estimate only the density derivatives, avoiding the computational burden of estimating the entire density function. By exploiting the landscape of the KDE, mean-shift methods offer a simpler and more efficient alternative to the previous methods. However, the choice of bandwidth crucial for the performance of mean-shift and density-based algorithms. An inappropriate bandwidth can introduce significant bias or cause the algorithm to converge to local modes, similar to many gradient-based optimization methods, ultimately hindering the detection of global modes. While considerable work has been done on bandwidth selection (Comaniciu, 2003; Chacón and Monfort, 2014) and adaptive bandwidth strategies (Chen et al., 2008; Zhao et al., 2009), mean-shift algorithms still lack inherent mechanisms for simultaneously exploring the search space to solve multimodal estimation problems.

Recently, the natural variational annealing (NVA) framework (Le Minh et al., 2025) has emerged as a gradient-based optimization method based on variational approximations of Gibbs measures (3). By temperature annealing, NVA effectively controls the sharpness

of the Gibbs measures, regulating the exploration-exploitation trade-off. The flexibility of the variational search distribution makes NVA particularly well-suited to multimodal optimization. Notably, the use of mixtures enables the simultaneous tracking of multiple modes, offering a robust solution to the multimodal estimation problem.

In this paper, we start by recalling the main principles of the NVA framework. Then, we extend it to mode estimation from probability distribution samples, combining the flexibility of variational search distributions with the computational efficiency of gradient-based methods. The resulting algorithm is called *empirical* NVA (eNVA). We show that Gaussian mean-shift can be viewed as a special case of eNVA with an adaptive learning rate, where the variational family consists of fixed-covariance Gaussian distributions. When the covariance is not fixed, Gaussian eNVA can therefore be viewed as a mean-shift algorithm with adaptive bandwidth. Finally, we further exploit NVA's flexibility to derive a multimodal optimization algorithm based on Gaussian mixtures (eNVA-GM), highlighting its potential for identifying multiple modes efficiently.

2 Natural variational annealing

In this section, we recall the key concepts of the Natural Variational Annealing (NVA) framework for global optimization (Le Minh et al., 2025), which forms the basis of our method. The NVA framework integrates three key concepts: variational optimization, entropy annealing, and natural gradients. We will build the NVA framework from scratch, explaining each concept sequentially. While our objective is to develop an optimization algorithm to locate the mode of a function p, this section considers a general smooth function f. We assume that f has a finite number of modes $\{\boldsymbol{x}_1^*, \dots, \boldsymbol{x}_I^*\}$.

2.1 Principle of variational optimization

Variational optimization is based on the lower bound for the global maximum:

$$\mathbb{E}_q[f(\boldsymbol{X})] \leq \max_{\boldsymbol{x}\in\mathbb{R}^d} f(\boldsymbol{x}).$$

where the expectation is taken with respect to a probability distribution q. The *variational formulation* of this problem rewrites it as maximizing this lower bound on the space of all probability distribution $\mathcal{P}(\mathbb{R}^d)$ on $\mathbb{R}^d$, leading to

$$q^* \in \arg\max_{q\in\mathcal{P}(\mathbb{R}^d)} \mathbb{E}_q[f(\boldsymbol{X})]. \tag{1}$$

The solutions of problem (1) are directly related to the maximum of f, since they take the form $\sum_{i=1}^I c_i \delta_{\boldsymbol{x}_i^*}$, where $(\boldsymbol{x}_i^*)_{i\in[I]}$ are the modes of f, and $(c_i)_{i\in[I]}$ are coefficients summing to 1.

However, this solution is singular. This motivates the introduction of an entropy penalty term, which yields the modified optimization problem:

$$q^{*,\omega} = \arg\max_{q \in \mathcal{P}(\mathbb{R}^d)} \mathbb{E}_q[f(\boldsymbol{X})] + \omega \mathcal{H}(q), \tag{2}$$

where $\omega > 0$ is a regularization parameter and $\mathcal{H}(q) := -\mathbb{E}_q[\log q(\boldsymbol{X})]$ denotes the entropy of q. Unlike the original formulation, this penalized problem is strictly convex and admits a unique solution, given by the Gibbs measure g_ω defined by

$$q^{*,\omega}(\boldsymbol{X}) = g_\omega(\boldsymbol{X}) := \frac{\exp(f(\boldsymbol{X})/\omega)}{\int \exp(f(\boldsymbol{X})/\omega) d\boldsymbol{X}}. \tag{3}$$

The parameter ω, often referred to as the *temperature* by analogy with statistical physics, governs the shape of the Gibbs measure. Gibbs measures exhibit two crucial properties: they reflect the variations of f, and as $\omega \to 0$, they concentrate on the global modes of f.

2.2 Natural variational family and natural gradient optimization

To solve problem (2) for a fixed ω, the search space for the distribution q is often restricted to a parameterized family, called the *variational family*. Let $q_{\boldsymbol{\lambda}}$ be a distribution parameterized by a vector $\boldsymbol{\lambda}$. The constrained problem can be written as:

$$\boldsymbol{\lambda}^{*,\omega} \in \arg\max_{\boldsymbol{\lambda}} \mathbb{E}_{q_{\boldsymbol{\lambda}}}[f(\boldsymbol{X})] + \omega \mathcal{H}(q_{\boldsymbol{\lambda}}). \tag{4}$$

The solutions of this problem may not be unique, for any solution $\boldsymbol{\lambda}^{*,\omega}$, the resulting distribution $q_{\boldsymbol{\lambda}^{*,\omega}}$ is a *variational approximation* of $q^{*,\omega} = g_\omega$, minimizing $\mathrm{KL}(q_{\boldsymbol{\lambda}^{*,\omega}} || g_\omega)$.

For instance, consider the family of Gaussian distributions. A Gaussian density $q_{\boldsymbol{\lambda}}$ can be parameterized by its natural parameters $\boldsymbol{\lambda} = (\boldsymbol{S}\boldsymbol{\mu}, -\boldsymbol{S}/2)$, where $\boldsymbol{\mu}$ is the mean and $\boldsymbol{S}$ is the precision matrix. This transforms the original problem into the maximization of the objective function $\mathcal{L}_\omega(\boldsymbol{\lambda}) := \mathbb{E}_{q_{\boldsymbol{\lambda}}}[f(\boldsymbol{X})] + \omega \mathcal{H}(q_{\boldsymbol{\lambda}})$, over the natural parameters of the Gaussian distribution.

While simple gradient ascent procedures can optimize $\mathcal{L}_\omega$, natural gradient ascent is more efficient (Amari, 1998). Natural gradients account for the geometry of the parameter space, indicating the direction of steepest ascent in the Riemannian manifold induced by the Fisher Information Matrix (FIM) (Bonnabel, 2013). This makes natural gradients independent from the parameterization and generally leads to faster convergence (Sato, 2001; Honkela et al., 2007). Natural gradient of $\mathcal{L}_\omega$ is:

$$\tilde{\nabla}_{\boldsymbol{\lambda}} \mathcal{L}_\omega(\boldsymbol{\lambda}) = \boldsymbol{F}(\boldsymbol{\lambda})^{-1} \nabla_{\boldsymbol{\lambda}} \mathcal{L}_\omega(\boldsymbol{\lambda}),$$

where $\boldsymbol{F}$ is the FIM. The natural gradient ascent update rule is then:

$$\boldsymbol{\lambda}_{t+1} = \boldsymbol{\lambda}_t + \rho_t \tilde{\nabla}_{\boldsymbol{\lambda}} \mathcal{L}_\omega(\boldsymbol{\lambda})|_{\boldsymbol{\lambda}_t}, \tag{5}$$

where $\rho_t > 0$ is the learning rate (or step size) at iteration t. The usual difficulty in using natural gradients is the estimation of the FIM and its inversion, which are costly or unfeasible in general. However, for exponential family distributions, the computation of natural gradients with respect to natural parameters does not require the estimation of the FIM and its inversion. Indeed, these natural gradients correspond to vanilla gradients computed with respect to the expectation parameters $\boldsymbol{M} = \mathbb{E}[\boldsymbol{T}(\boldsymbol{X})]$, where $\boldsymbol{T}(\boldsymbol{X})$ denotes the sufficient statistics:

$$\tilde{\nabla}_{\boldsymbol{\lambda}} \mathcal{L}_\omega(\boldsymbol{\lambda}) = \nabla_{\boldsymbol{M}} \mathcal{L}_\omega(\boldsymbol{\lambda}). \tag{6}$$

In this paper, we are going to consider three variational families: 1) Gaussian distributions with fixed covariance, 2) Gaussian distributions, and 3) Gaussian mixtures. While Gaussian mixtures do not belong to the exponential family, they belong to the more general *minimal conditional exponential family* (MCEF, Lin et al., 2019), for which natural parameters can be defined and the natural gradients can be computed like for regular exponential family distributions.

2.3 Entropy annealing

The process of controlling the temperature ω is known as *annealing*. As stated previously, decreasing ω concentrates the Gibbs measure g_ω, therefore helping with identifying the global modes of f in a global optimization problem.

The temperature can be controlled through a sequence $(\omega_t)_{t\geq 1}$, called *annealing schedule*, setting the value of ω at each iteration. Therefore, the update rule (5) is modified to:

$$\boldsymbol{\lambda}_{t+1} = \boldsymbol{\lambda}_t + \rho_t \tilde{\nabla}_{\boldsymbol{\lambda}} \mathcal{L}_{\omega_t}(\boldsymbol{\lambda})|_{\boldsymbol{\lambda}_t}. \tag{7}$$

This is the NVA update rule.

At high ω, the Gibbs measure is smooth, promoting diversity by increasing the solution's spread. As $\omega \to 0$, the Gibbs measure concentrates, and therefore the solution also concentrates. For Gaussian distributions, the mean can be seen as a particle moving in the search space. In this case, the temperature ω modulates the exploration of the search space by the particle. Because $\omega = 0$ leads to a degenerate solution where the covariance matrix vanishes, $\omega \to 0$ also ensures that the mean converges to a mode.

Behavior of the Gaussian variational solution. To explain this, we remark that with a fixed ω, Gaussian solutions with parameters $(\boldsymbol{\mu}^{*,\omega}, (\boldsymbol{S}^{*,\omega})^{-1})$ must satisfy

$$\begin{cases} \nabla_{\boldsymbol{\mu}} \mathrm{KL}(q_{\boldsymbol{\lambda}} \mid\mid g_\omega)|_{\boldsymbol{\lambda}^{*,\omega}} = 0 \\ \nabla_{\boldsymbol{S}^{-1}} \mathrm{KL}(q_{\boldsymbol{\lambda}} \mid\mid g_\omega)|_{\boldsymbol{\lambda}^{*,\omega}} = 0 \end{cases}, \tag{8}$$

where g_ω is the Gibbs measure defined by (3) and $\boldsymbol{\lambda}^{*,\omega} = (\boldsymbol{S}^{*,\omega}\boldsymbol{\mu}^{*,\omega}, -\boldsymbol{S}^{*,\omega}/2)$.

If the covariance matrix is fixed, only the condition $\nabla_{\boldsymbol{\mu}} \mathrm{KL}(q_{\boldsymbol{\lambda}} \mid\mid g_\omega)|_{\boldsymbol{\lambda}^{*,\omega}} = 0$ is needed. We have

$$\nabla_{\boldsymbol{\mu}} \mathrm{KL}(q_{\boldsymbol{\lambda}} \mid\mid g_\omega)|_{\boldsymbol{\lambda}^{*,\omega}} = \mathbb{E}_{\mathcal{N}(\boldsymbol{\mu}^{*,\omega},(\boldsymbol{S}^{*,\omega})^{-1})} \left[\nabla_{\boldsymbol{\xi}} f(\boldsymbol{\xi})\right] = 0, \tag{9}$$

so this implies possible solutions $\boldsymbol{\mu}^{*,\omega}$ are close to a global mode $\boldsymbol{x}_i^*$ of f.

When the covariance matrix is not fixed, the condition $\nabla_{\boldsymbol{S}^{-1}} \mathrm{KL}(q_{\boldsymbol{\lambda}} \mid\mid g_\omega)|_{\boldsymbol{\lambda}^{*,\omega}} = 0$ is also needed. Since $\boldsymbol{\mu}^{*,\omega}$ remains in a neighborhood of $\boldsymbol{x}_i^*$, This condition implies

$$(\boldsymbol{S}^{*,\omega})^{-1} = -\omega \mathbb{E}_{\mathcal{N}(\boldsymbol{\mu}^{*,\omega},(\boldsymbol{S}^{*,\omega})^{-1})} \left[\nabla^2_{\boldsymbol{\xi}} f(\boldsymbol{\xi})\right]^{-1} \xrightarrow[\omega \to 0]{} 0. \tag{10}$$

Therefore, when the temperature decreases to 0, the optimal covariance matrices vanish to 0 and the mean converges to a (global or local) mode of f.

Importance of the annealing schedule. It is natural to wonder why one should use an entropy penalty and annealing instead of directly solving the variational problem for $\omega = 0$, which is needed for convergence. The key reason lies in the role of the Gaussian distribution's spread, which determines the particle's ability to explore the search space effectively. The Gaussian mean, viewed as a particle, scans the landscape in its neighborhood at each iteration and stays at the "mean" of the scanned area. Meanwhile, the Gaussian covariance matrix acts as the scanning range, as described by condition (9). As there can be as many local solutions satisfying (8) as the number of local modes of f, a particle unable to reach a global mode may converge to a closer local one.

When ω is small, the Gibbs measure sharpens, narrowing the spread of the Gaussian distribution. This makes the particle more likely to converge to the closest mode, regardless of whether it is a local or global one. In contrast, a larger ω smoothens the Gibbs measure, broadening the Gaussian's spread. This allows the particle to scan larger regions of the search space and detect global modes farther away. After sufficient exploration, ω can be gradually decreased to enable the particle to converge to the highest-quality mode within its reach.

This shows how the annealing schedule $(\omega_t)_{t\geq 1}$ should be set. The initial temperature must be large enough and should decrease slowly so that the Gaussian spread covers the desired mode during the whole run. At the end of the run, the temperature should be small enough to allow convergence of the particle to the mode. The optimal schedule is problem-dependent, so trial-and-error can be used to choose one.

3 Applying NVA on samples: empirical NVA

In the previous section, we have introduced the NVA framework for global optimization. Here, we describe its application to mode-estimation problems using a finite sample from p. We call this *empirical NVA* (eNVA). In this section, we assume that p has a unique mode $\boldsymbol{x}^*$. The mode-estimation problem can be formulated as constructing a consistent

estimator for $\boldsymbol{x}^*$. We derive an algorithm based a Gaussian variational family, initially with a fixed covariance matrix, and later in the general case. We also show how mean-shift emerges as a special case of fixed-covariance Gaussian eNVA.

3.1 Fixed-covariance Gaussian eNVA

First, we investigate the variational family formed by the Gaussian distributions $q_{\boldsymbol{\lambda}}$ with fixed covariance $\boldsymbol{S}^{-1} = s^{-1}\boldsymbol{I}$, parameterized by its natural parameter $\boldsymbol{\lambda} = s\boldsymbol{\mu}$. The NVA update rule derived from (7) and (6) is:

$$\boldsymbol{\mu}_{t+1} = \boldsymbol{\mu}_t + \rho_t s^{-1} \nabla_{\boldsymbol{\mu}} \mathcal{L}_{\omega_t}(\boldsymbol{\lambda})|_{\boldsymbol{\lambda}_t},$$

where $\mathcal{L}_\omega(\boldsymbol{\lambda}) = \mathcal{L}_0(\boldsymbol{\lambda}) + \omega\mathcal{H}(q_{\boldsymbol{\lambda}})$ and $\mathcal{L}_0(\boldsymbol{\lambda}) = \mathbb{E}_{q_{\boldsymbol{\lambda}}}[p(\boldsymbol{X})]$. To implement the update rule, the gradient $\nabla_{\boldsymbol{\mu}} \mathcal{L}_\omega(\boldsymbol{\lambda})|_{\boldsymbol{\lambda}_t}$ must be computed.

Gradient expression. We define

$$\nabla_{\boldsymbol{\mu}} \mathcal{L}_\omega(\boldsymbol{\lambda}) = \boldsymbol{g}(\boldsymbol{\lambda}) + \omega\boldsymbol{\gamma}(\boldsymbol{\lambda}), \tag{11}$$

where $\boldsymbol{g}(\boldsymbol{\lambda}) := \nabla_{\boldsymbol{\mu}} \mathbb{E}_{q_{\boldsymbol{\lambda}}}[p(\boldsymbol{X})]$ and $\boldsymbol{\gamma}(\boldsymbol{\lambda}) := \nabla_{\boldsymbol{\mu}} \mathcal{H}(q_{\boldsymbol{\lambda}})$.

Because $q_{\boldsymbol{\lambda}}$ is a Gaussian distribution, we have $\boldsymbol{\gamma}(\boldsymbol{\lambda}) = 0$. For $\boldsymbol{g}(\boldsymbol{\lambda})$, we have

$$\boldsymbol{g}(\boldsymbol{\lambda}) = s\mathbb{E}_{q_{\boldsymbol{\lambda}}}[(\boldsymbol{X} - \boldsymbol{\mu})p(\boldsymbol{X})]. \tag{12}$$

If p can be evaluated, then these gradients can be estimated through Monte Carlo approximations by sampling $\boldsymbol{X}$ under $q_{\boldsymbol{\lambda}}$, but in our case, p is unavailable.

Alternatively, based on the fact that p is a probability distribution, we can write

$$\boldsymbol{g}(\boldsymbol{\lambda}) = s\mathbb{E}_p[(\boldsymbol{X} - \boldsymbol{\mu})q_{\boldsymbol{\lambda}}(\boldsymbol{X})].$$

Again, if we can sample from p, then we can also use a Monte Carlo approximation to estimate $\boldsymbol{g}(\boldsymbol{\lambda})$. However, in our problem, we cannot directly sample from p, as we only have a fixed sample $\boldsymbol{X}_{1:N} = (\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N)$ with distribution p.

Empirical approximation. To circumvent this issue, we remark that $\mathcal{L}_0(\boldsymbol{\lambda})$ can also be written

$$\mathcal{L}_0(\boldsymbol{\lambda}) = \mathbb{E}_p[q_{\boldsymbol{\lambda}}(\boldsymbol{X})].$$

Therefore, it can be empirically approximated by

$$\widehat{\mathcal{L}}_{0,N}(\boldsymbol{\lambda}) = \frac{1}{N}\sum_{i=1}^{N} q_{\boldsymbol{\lambda}}(\boldsymbol{X}_i) = \mathbb{E}_{p_N}[q_{\boldsymbol{\lambda}}(\boldsymbol{X})],$$

where p_N is the empirical measure $N^{-1}\sum_{i=1}^N \delta_{\boldsymbol{X}_i}$. The approximation $\widehat{\mathcal{L}}_{0,N}(\boldsymbol{\lambda})$ is in fact the kernel density estimate (KDE) of p, with Gaussian kernel and bandwidth s^{-1}. Since $\boldsymbol{X}_{1:N}$ is a i.i.d. sequence, the function $\mathcal{L}$ converges uniformly on all compact sets of $\mathbb{R}^d$.

However, the KDE is biased and the consistency of $\widehat{\boldsymbol{\mu}}_N = \widehat{\boldsymbol{\lambda}}_N = \arg\max \widehat{\mathcal{L}}_{0,N}(\boldsymbol{\lambda})$ can only be obtained if the bandwidth $s^{-1/2} \to 0$. To derive sufficient conditions for consistency, we define a sequence $(s_N)_{N\geq 1}$, modifying the Gaussian covariance matrix with the sample size N. The following theorem gives sufficient conditions on s_N for the consistency of $\widehat{\boldsymbol{\mu}}_N$ as an estimator for $\boldsymbol{x}^*$.

Theorem 3.1. *If* $s_N^{-1} \xrightarrow[N\to\infty]{} 0$ *and* $Ns_N^{-1} \xrightarrow[N\to\infty]{} +\infty$*, then we have* $\widehat{\boldsymbol{\mu}}_N \xrightarrow[N\to\infty]{\mathbb{P}} \boldsymbol{x}^*$.

Proof. $\widehat{\mathcal{L}}_{0,N}(\boldsymbol{\lambda})$ is a kernel density estimate with Gaussian kernel, bandwidth $s_N^{-1/2}$, based on the samples $X_1, \dots, X_N$. This result is a multivariate generalization of Theorem 3A of Parzen (1962) on the consistency of the modes of kernel density estimates, ensuring that the maximizers of $\mathcal{L}_{0,N}(\boldsymbol{\lambda})$ are consistent for $\boldsymbol{x}^*$ if $s_N^{-1/2} \xrightarrow[N\to\infty]{} 0$ and $Ns_N^{-1} \xrightarrow[N\to\infty]{} +\infty$. □

In practice, N is fixed and $X_{1:n}$ is a sample of fixed size, so the result means that s^{-1} should neither be too large nor too small to obtain a good estimator for $\boldsymbol{x}^*$.

Gradient estimation. The empirical NVA (eNVA) framework consists in optimizing $\widehat{\mathcal{L}}_{0,N}$ instead of $\mathcal{L}_0$. This approximation introduces an error that vanishes when N becomes large. In fixed-covariance Gaussian eNVA, the natural gradient can be estimated by

$$\widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:N};\boldsymbol{\lambda}) := \nabla_\mu \widehat{\mathcal{L}}_{0,N}(\boldsymbol{\lambda}) = \frac{1}{N}\sum_{i=1}^N s(\boldsymbol{X}_i - \boldsymbol{\mu}) q_{\boldsymbol{\lambda}}(\boldsymbol{X}_i), \tag{13}$$

Finally, the following update rule can be used to implement a fixed-covariance Gaussian eNVA algorithm:

$$\boldsymbol{\mu}_{t+1} = \boldsymbol{\mu}_t + \rho_t s^{-1} \widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:N};\boldsymbol{\lambda}_t). \tag{14}$$

When $\widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:N};\boldsymbol{\lambda}_t)$ is too costly to compute because N is too large, we can approximate this estimator using B i.i.d. samples of p_N, $B \leq N$, denoted $\boldsymbol{X}_{1:B}^{(b)}$. Sampling from p_N means sampling uniformly with replacement in the elements of $\boldsymbol{X}_{1:N}$. Finally,

$$\widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:B}^{(b)};\boldsymbol{\lambda}) = \frac{1}{B}\sum_{i=1}^B s(\boldsymbol{X}_i^{(b)} - \boldsymbol{\mu}) q_{\boldsymbol{\lambda}}(\boldsymbol{X}_i^{(b)}),$$

is unbiased for $\nabla_{\boldsymbol{\mu}} \widehat{\mathcal{L}}_{0,N}(\boldsymbol{\lambda})$, which ensure consistency of the natural gradient ascent.

Algorithm. In fixed-covariance Gaussian eNVA, the entropy penalty has no effect, because it does not depend on the mean of the Gaussian, which is the only parameter to optimize. Algorithm 1 gives an implementation when N is large and $\widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:N}; \boldsymbol{\lambda})$ is too costly to compute, therefore using the Monte Carlo approximation $\widehat{\boldsymbol{g}}(\boldsymbol{X}^{(b)}_{1:B}; \boldsymbol{\lambda})$ instead, as described above. The hyperparameters are the number of iterations T, the number of samples B used to compute the gradients, the covariance matrix $s^{-1}\boldsymbol{I}$ of the Gaussian distributions, the initial parameters $\boldsymbol{\theta}_0 = \boldsymbol{\mu}_0$, and the learning rate schedule $(\rho_t)_{t\in[T]}$.

Algorithm 1: Fixed-covariance Gaussian eNVA

```
Given samples X_{1:n}.
Set T, B, s, θ_0, (ρ_t)_{t∈[T]}.
for t = 0:(T − 1) do
    Sample X_i^{(b)} ~ i.i.d. U({X_1, ..., X_n}),   for i = 1:B.
    Compute ĝ(X_{1:B}^{(b)}; λ_t).
    Update μ_{t+1} = μ_t + ρ_t s^{-1} ĝ(X_{1:B}^{(b)}; λ_t).
end
return θ_T.
```

3.2 Link to Gaussian mean-shift

The mean-shift procedure is a mode-finding algorithm that estimates the modes of a distribution by performing gradient ascent on the KDE with a specified kernel and bandwidth (Fukunaga and Hostetler, 1975; Carreira-Perpiñan, 2007). Here, we demonstrate that the mean-shift algorithm with a Gaussian kernel is a particular instance of the Gaussian eNVA with a fixed covariance matrix.

The KDE for the samples $\boldsymbol{X}_{1:N}$, using Gaussian kernel and bandwidth $s^{-1/2}$, is given by

$$\widehat{p}(\boldsymbol{\xi}) = \frac{1}{N}\sum_{i=1}^{N} q_{\boldsymbol{\theta}_i}(\boldsymbol{\xi}),$$

where $q_{\boldsymbol{\theta}_i}$ denotes the Gaussian density with natural parameters $\boldsymbol{\theta}_i = (s\boldsymbol{X}_i, -s/2)$. The mean-shift algorithm aims to find the stationary points of $\widehat{p}$, which means those satisfying

$$\nabla\widehat{p}(\boldsymbol{\xi}) = -\frac{1}{N}\sum_{i=1}^{N} s(\boldsymbol{\xi} - \boldsymbol{X}_i) q_{\boldsymbol{\theta}_i}(\boldsymbol{\xi}) = 0.$$

To solve this, a fixed-point iteration is used to update the position of a particle $\boldsymbol{\mu}_{t+1} = f(\boldsymbol{\mu}_t)$ where

$$f(\boldsymbol{\mu}) = \sum_{i=1}^{N} \frac{q_{\boldsymbol{\theta}_i}(\boldsymbol{\mu})}{\sum_{j=1}^{N} q_{\boldsymbol{\theta}_j}(\boldsymbol{\mu})} \boldsymbol{X}_i.$$

The update rule can then be written as

$$\begin{aligned}\boldsymbol{\mu}_{t+1} &= \boldsymbol{\mu}_t + \sum_{i=1}^{N} \frac{q_{\boldsymbol{\theta}_i}(\boldsymbol{\mu}_t)}{\sum_{j=1}^{N} q_{\boldsymbol{\theta}_j}(\boldsymbol{\mu}_t)}(\boldsymbol{X}_i - \boldsymbol{\mu}_t),\\ &= \boldsymbol{\mu}_t + \rho(\boldsymbol{\lambda}_t) s^{-1} \widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:n}; \boldsymbol{\lambda}_t),\\ &= \boldsymbol{\mu}_t + \rho(\boldsymbol{\lambda}_t) s^{-1} \nabla_{\boldsymbol{\mu}} \widehat{\mathcal{L}}_{\omega,N}(\boldsymbol{\lambda}_t),\end{aligned}$$

where $\boldsymbol{\lambda}_t = (s\boldsymbol{\mu}_t, -s/2)$, $\rho(\boldsymbol{\lambda}) = (\sum_{j=1}^{N} q_{\boldsymbol{\lambda}}(\boldsymbol{X}_j))^{-1}$ defines an adaptive step schedule and $\widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:n}; \boldsymbol{\lambda}_t)$ is defined as in (13).

This update is nearly identical to the fixed-covariance eNVA update (14), with the sole difference being the adaptive learning rate $\rho(\boldsymbol{\lambda}_t)$. Therefore, Gaussian mean-shift can be regarded as a special case of Gaussian eNVA. The bandwidth parameter in mean-shift plays the same role as the covariance matrix in eNVA, and both share the same interpretation.

The behavior of both algorithms is very sensitive to the selection of s^{-1}, which controls the inherent error due to the KDE. In the eNVA context, large s^{-1} implies a particle with a broader range; in mean-shift, it leads to a smoother KDE landscape, potentially masking the undesired local modes. However, this comes at the cost of increased bias, as the solution $\boldsymbol{\mu}_\infty$ may deviate from the true mode $\boldsymbol{x}^*$. Similarly, a small s^{-1} results in a narrower particle range in eNVA or a sharper KDE landscape in mean-shift, resulting in convergence to undesired modes. An optimal value of s^{-1} exists that balances this bias-variance trade-off, minimizing the bias while avoiding overfitting to local structures.

Now, we will see that the NVA framework can be used to automatically adjust s^{-1} during the algorithm. Removing the fixed-covariance constraint in Gaussian NVA enables the covariance matrix to adapt via natural gradient ascent. In the mean-shift analogy, this means that the bandwidth of the KDE is adaptive. This will enhance mode-estimation performance, as starting with large s^{-1} and progressively reducing it as the algorithm converges could favor convergence to the correct modes. We investigate this direction in the next section, where we derive a general Gaussian eNVA algorithm.

3.3 General Gaussian eNVA

When $\boldsymbol{S}$ is not fixed, the Gaussian NVA update rules are:

$$\begin{aligned}\boldsymbol{S}_{t+1} &= \boldsymbol{S}_t - 2\rho_t \nabla_{\boldsymbol{S}^{-1}} \mathcal{L}_\omega(\boldsymbol{\lambda})|_{\boldsymbol{\lambda}_t},\\ \boldsymbol{\mu}_{t+1} &= \boldsymbol{\mu}_t + \rho_t \boldsymbol{S}_{t+1}^{-1} \nabla_{\boldsymbol{\mu}} \mathcal{L}_\omega(\boldsymbol{\lambda})|_{\boldsymbol{\lambda}_t},\end{aligned}$$

where

$$\begin{aligned}\nabla_{\boldsymbol{\mu}} \mathcal{L}_\omega(\boldsymbol{\lambda}) &= \mathbb{E}_{q_{\boldsymbol{\lambda}}}[\boldsymbol{S}(\boldsymbol{X} - \boldsymbol{\mu})\mathcal{L}_\omega(\boldsymbol{\lambda})],\\ \nabla_{\boldsymbol{S}^{-1}} \mathcal{L}_\omega(\boldsymbol{\lambda}) &= \mathbb{E}_{q_{\boldsymbol{\lambda}}}[(\boldsymbol{S}(\boldsymbol{X} - \boldsymbol{\mu})(\boldsymbol{X} - \boldsymbol{\mu})^T \boldsymbol{S} - \boldsymbol{S})\mathcal{L}_\omega(\boldsymbol{\lambda})].\end{aligned}$$

To derive an algorithm, we need to compute gradients of the form $\nabla_{\boldsymbol{\mu}} \mathcal{L}_\omega(\boldsymbol{\lambda})$ and $\nabla_{\boldsymbol{S}^{-1}} \mathcal{L}_\omega(\boldsymbol{\lambda})$.

Gradient expressions. In addition to (11), we define

$$\nabla_{\boldsymbol{S}^{-1}}\mathcal{L}_\omega(\boldsymbol{\lambda}) = \boldsymbol{H}(\boldsymbol{\lambda}) + \omega\boldsymbol{\eta}(\boldsymbol{\lambda}),$$

where $\boldsymbol{H}(\boldsymbol{\lambda}) := \nabla_{\boldsymbol{S}^{-1}}\mathbb{E}_{q_{\boldsymbol{\lambda}}}[p(\boldsymbol{X})]$ and $\boldsymbol{\eta}(\boldsymbol{\lambda}) := \nabla_{\boldsymbol{S}^{-1}}\mathcal{H}(q_{\boldsymbol{\lambda}})$.

Because $q_{\boldsymbol{\lambda}}$ is a Gaussian distribution, the gradient $\boldsymbol{\gamma}(\boldsymbol{\lambda}) = 0$ remains unchanged, and $\boldsymbol{\eta}(\boldsymbol{\lambda}) = \boldsymbol{S}/2$. The expressions of $\boldsymbol{g}(\boldsymbol{\lambda})$ and $\boldsymbol{H}(\boldsymbol{\lambda})$ are

$$\begin{aligned}\boldsymbol{g}(\boldsymbol{\lambda}) &= \mathbb{E}_{q_{\boldsymbol{\lambda}}}[\boldsymbol{S}(\boldsymbol{X}-\boldsymbol{\mu})p(\boldsymbol{X})],\\ \boldsymbol{H}(\boldsymbol{\lambda}) &= \frac{1}{2}\mathbb{E}_{q_{\boldsymbol{\lambda}}}[(\boldsymbol{S}(\boldsymbol{X}-\boldsymbol{\mu})(\boldsymbol{X}-\boldsymbol{\mu})^T\boldsymbol{S}-\boldsymbol{S})p(\boldsymbol{X})].\end{aligned}$$

This yields

$$\begin{aligned}\boldsymbol{g}(\boldsymbol{\lambda}) &= \mathbb{E}_{p}[\boldsymbol{S}(\boldsymbol{X}-\boldsymbol{\mu})q_{\boldsymbol{\lambda}}(\boldsymbol{X})],\\ \boldsymbol{H}(\boldsymbol{\lambda}) &= \frac{1}{2}\mathbb{E}_{p}[(\boldsymbol{S}(\boldsymbol{X}-\boldsymbol{\mu})(\boldsymbol{X}-\boldsymbol{\mu})^T\boldsymbol{S}-\boldsymbol{S})q_{\boldsymbol{\lambda}}(\boldsymbol{X})].\end{aligned}$$

Empirical approximation. As before, we use the samples to construct the empirical approximation $\widehat{\mathcal{L}}_{0,N}(\boldsymbol{\lambda}) = \mathbb{E}_{p_N}[q_{\boldsymbol{\lambda}}(\boldsymbol{X})]$. However, when the covariance is not fixed, the function $\widehat{\mathcal{L}}_{0,N}(\boldsymbol{\lambda})$ cannot be maximized over $\boldsymbol{\Theta}$, the set of natural parameters of the form $\boldsymbol{\lambda} = (\boldsymbol{S}\mu, -\boldsymbol{S}/2)$. This is because $\boldsymbol{\Theta}$ is not compact and $\widehat{\mathcal{L}}_{0,N}(\boldsymbol{\lambda})$ diverges to $+\infty$ as $\boldsymbol{\mu} \to \boldsymbol{X}_i$, for any $1 \le i \le N$, and $\boldsymbol{S}^{-1} \to 0$. Consequently, unlike in the fixed covariance case, there is no guarantee of obtaining a consistent estimator because $\boldsymbol{S}$ is unconstrained. To address this issue in practice, we can impose a lower bound on the eigenvalues of the covariance matrix by applying a damping correction after the natural gradient update:

$$\boldsymbol{S}_{t+1} \to (\boldsymbol{S}_{t+1}^{-1} + \alpha_N\boldsymbol{I})^{-1},$$

where $\alpha_N > 0$. This correction ensures that $\boldsymbol{S}_{t+1}^{-1} \ge \alpha_N\boldsymbol{I}$. Whereas this damping strategy departs from the NVA framework, it prevents the covariance matrix from becoming too small. By doing so, $\widehat{\mathcal{L}}_{0,N}(\boldsymbol{\lambda})$ remains bounded. When α_N is small, the Gaussian is concentrated near the mode $\boldsymbol{x}^*$, which means $\boldsymbol{S}^{-1}$ is close its lower bound α_N. Therefore, sufficient conditions for consistency may be similar to the fixed-covariance case, i.e. $\alpha_N^{1/2} \xrightarrow[N\to\infty]{} 0$ and $N\alpha_N \xrightarrow[N\to\infty]{} +\infty$, but the proof of such result is out of the scope of this paper.

Algorithm. Algorithm 2 gives an implementation for Gaussian eNVA when N is large and $\widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:N};\boldsymbol{\lambda})$ and $\widehat{\boldsymbol{H}}(\boldsymbol{X}_{1:N};\boldsymbol{\lambda})$ are too costly to compute, therefore using the Monte Carlo approximations $\widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:B}^{(b)};\boldsymbol{\lambda})$ and $\widehat{\boldsymbol{H}}(\boldsymbol{X}_{1:B}^{(b)};\boldsymbol{\lambda})$ instead, where $\boldsymbol{X}_{1:B}^{(b)}$ is a sequence of

size B consisting of i.i.d. elements sampled from $\{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N\}$, and

$$\widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:B}^{(b)}; \boldsymbol{\lambda}) = \frac{1}{B} \sum_{i=1}^{B} \boldsymbol{S}(\boldsymbol{X} - \boldsymbol{\mu}) q_{\boldsymbol{\lambda}}(\boldsymbol{X}_i),$$

$$\widehat{\boldsymbol{H}}(\boldsymbol{X}_{1:B}^{(b)}; \boldsymbol{\lambda}) = \frac{1}{2B} \sum_{i=1}^{B} (\boldsymbol{S}(\boldsymbol{X} - \boldsymbol{\mu})(\boldsymbol{X} - \boldsymbol{\mu})^T \boldsymbol{S} - \boldsymbol{S}) q_{\boldsymbol{\lambda}}(\boldsymbol{X}_i),$$

are unbiased estimators of $\nabla_{\boldsymbol{\mu}} \widehat{\mathcal{L}}_{0,N}(\boldsymbol{\lambda})$ and $\nabla_{\boldsymbol{S}^{-1}} \widehat{\mathcal{L}}_{0,N}(\boldsymbol{\lambda})$.

The hyperparameters are the number of iterations T, the number of samples B used to compute the gradients, the initial parameters $\boldsymbol{\theta}_0 = (\boldsymbol{\mu}_0, \boldsymbol{S}_0)$, the learning rate schedule $(\rho_t)_{t \in [T]}$, the learning rate schedule $(\omega_t)_{t \in [T]}$, and optionally, a lower bound for the covariance matrix eigenvalues α, called a *damping factor*.

Algorithm 2: Gaussian eNVA

Given samples $\boldsymbol{X}_{1:N}$.
Set T, B, $\boldsymbol{\theta}_0$, $(\rho_t)_{t \in [T]}$, $(\omega_t)_{t \in [T]}$, α (optional).
for $t = 0 : (T-1)$ **do**
  Sample $\boldsymbol{X}_i^{(b)} \overset{\text{i.i.d.}}{\sim} \mathcal{U}(\{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n\})$, for $i = 1 : B$.
  Compute $\widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:B}^{(b)}; \boldsymbol{\lambda}_t)$, $\widehat{\boldsymbol{H}}(\boldsymbol{X}_{1:B}^{(b)}; \boldsymbol{\lambda}_t)$.
  Update
    $\boldsymbol{S}_{t+1} = (1 - \omega_t \rho_t) \boldsymbol{S}_t - 2\rho_t \widehat{\boldsymbol{H}}(\boldsymbol{X}_{1:B}^{(b)}; \boldsymbol{\lambda}_t)$,
    $\boldsymbol{S}_{t+1} = (\boldsymbol{S}_{t+1}^{-1} + \alpha \boldsymbol{I})^{-1}$, (optional)
    $\boldsymbol{\mu}_{t+1} = \boldsymbol{\mu}_t + \rho_t \boldsymbol{S}_{t+1}^{-1} \widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:B}^{(b)}; \boldsymbol{\lambda}_t)$.
end
return $\boldsymbol{\theta}_T$.

Interplay between the annealing schedule and the damping factor. In fixed-covariance Gaussian eNVA, the entropy penalty has no effect. However, in general Gaussian eNVA, it plays a role in slowing the decay of $\boldsymbol{S}^{-1}$. Since $\boldsymbol{S}^{-1}$ controls the spread of the Gaussian, the annealing schedule can be adjusted to maintain a large exploration range for longer, by choosing a schedule that decays more slowly.

In the NVA theory, the annealing schedule ω_t should converge to 0 to ensure that $\boldsymbol{\mu}_t$ converges to $\boldsymbol{x}^*$, canceling all bias as discussed in Section 2.3. However, because the mode of p is estimated from a finite sample, an error is introduced into the optimization process, and $\boldsymbol{\mu}_t$ will not converge exactly to $\boldsymbol{x}^*$, even as $\omega_t \to 0$. This error is analogous to the one caused by the KDE in fixed-covariance Gaussian eNVA. It can only be optimized by

ω_∞	0	> 0	0
α	0	0	> 0
$\boldsymbol{S}_\infty^{-1}$	0	$\approx -\omega_\infty(\nabla^2 p(\boldsymbol{x}^*))^{-1}$	$\approx \alpha \boldsymbol{I}$
KDE error	high	optimizable	optimizable

Table 1: The error due to the kernel density estimate on the estimation of the mode $\boldsymbol{x}^*$ of p can be optimized by adjusting the limiting covariance matrix $\boldsymbol{S}_\infty^{-1}$ using ω_∞ or α.

setting the limit $\boldsymbol{S}_\infty^{-1}$ of the covariance matrix $\boldsymbol{S}_t^{-1}$, by analogy with the bandwidth of the KDE.

The error can be optimized in two ways: by setting the damping factor α, or by specifying a positive limit ω_∞ for the annealing schedule ω_t, which eliminates the need for the damping step. These two approaches for controlling the KDE error are summarized in Table 1.

4 Multimodal estimation with empirical NVA

In the previous sections, we have shown how NVA can be applied for mode estimation of a sample using the eNVA algorithms and explored its connection with the well-known mean-shift algorithm. In general, NVA provides a flexible framework in global optimization through the choice of the variational family. This framework can be used for multimodal optimization by choosing Gaussian mixtures. Building on these concepts, we derive the eNVA updates for Gaussian mixtures to address multimodal estimation problems. In this section, we suppose that p admits multiple modes $\{\boldsymbol{x}_1^*, \dots, \boldsymbol{x}_I^*\}$.

4.1 Special case of Gaussian mixtures

Parameterization and update rule. We choose the Gaussian mixture family with a fixed number of components as the variational family. Although Gaussian mixtures are not exponential family distributions, they can be formulated as a *minimal conditional exponential family* (MCEF, Lin et al., 2019) with natural parameters. Specifically, they are expressed as

$$q_{\boldsymbol{\Lambda}} = \sum_{k=1}^{K} \pi_k q_{\boldsymbol{\lambda}_k},$$

where $(\pi_k)_{k\in[K]}$ are mixture weights summing to 1, $(q_{\boldsymbol{\lambda}_k})_{k\in[K]}$ are the Gaussian components with natural parameters, and

$$\boldsymbol{\Lambda} = (\log(\pi_1/\pi_K), \dots, \log(\pi_{K-1}/\pi_K), \boldsymbol{\lambda}_1, \dots, \boldsymbol{\lambda}_K)$$

are the natural parameters of the MCEF.

The NVA objective is given by $\mathcal{L}_\omega(\mathbf{\Lambda}) = \mathcal{L}_0(\mathbf{\Lambda}) + \omega\mathcal{H}(q_{\mathbf{\Lambda}})$, with $\mathcal{L}_0(\mathbf{\Lambda}) = \mathbb{E}_{q_{\mathbf{\Lambda}}}[p(\boldsymbol{X})]$. The update rule for the parameters of the mixtures is :

$$\mathbf{\Lambda}_{t+1} = \mathbf{\Lambda}_t + \rho_t \tilde{\nabla}_{\mathbf{\Lambda}} \mathcal{L}_{\omega_t}(\mathbf{\Lambda})|_{\mathbf{\Lambda}_t}. \tag{15}$$

By converting the natural gradient into a vanilla gradient with respect to the expectation parameters, and then using the chain rule, we obtain

$$\begin{aligned}
\boldsymbol{S}_{k,t+1} &= \boldsymbol{S}_{k,t} - \frac{2\rho_t}{\pi_{k,t}} \nabla_{\boldsymbol{S}_k^{-1}} \mathcal{L}_{\omega_t}(\mathbf{\Lambda})|_{\mathbf{\Lambda}_t}, \\
\boldsymbol{\mu}_{k,t+1} &= \boldsymbol{\mu}_{k,t} + \frac{\rho_t}{\pi_{k,t}} \boldsymbol{S}_{k,t+1}^{-1} \nabla_{\boldsymbol{\mu}_k} \mathcal{L}_{\omega_t}(\mathbf{\Lambda})|_{\mathbf{\Lambda}_t}, \\
v_{k,t+1} &= v_{k,t} + \rho_t \nabla_{\pi_k} \mathcal{L}_{\omega_t}(\mathbf{\Lambda})|_{\mathbf{\Lambda}_t},
\end{aligned}$$

where $v_k = \log(\pi_k/\pi_K)$, for $k \in [K-1]$.

Gradient computation and estimation. Like for Gaussian NVA, the gradients needed for Gaussian mixture NVA can be written as:

$$\begin{aligned}
\nabla_{\pi_k} \mathcal{L}_{\omega_t}(\mathbf{\Lambda}) &= f_k(\mathbf{\Lambda}) + \omega\varphi_k(\mathbf{\Lambda}), \\
\nabla_{\boldsymbol{\mu}_k} \mathcal{L}_{\omega_t}(\mathbf{\Lambda}) &= \boldsymbol{g}_k(\mathbf{\Lambda}) + \omega\boldsymbol{\gamma}_k(\mathbf{\Lambda}), \\
\nabla_{\boldsymbol{S}_k^{-1}} \mathcal{L}_{\omega_t}(\mathbf{\Lambda}) &= \boldsymbol{H}_k(\mathbf{\Lambda}) + \omega\boldsymbol{\eta}_k(\mathbf{\Lambda}),
\end{aligned}$$

where $f_k(\mathbf{\Lambda}) := \nabla_{\pi_k}\mathcal{L}_0(\mathbf{\Lambda})$, $\varphi_k(\mathbf{\Lambda}) := \nabla_{\pi_k}\mathcal{H}(q_{\mathbf{\Lambda}})$, $\boldsymbol{g}_k(\mathbf{\Lambda}) := \nabla_{\boldsymbol{\mu}_k}\mathcal{L}_0(\mathbf{\Lambda})$, $\boldsymbol{\gamma}_k(\mathbf{\Lambda}) := \nabla_{\boldsymbol{\mu}_k}\mathcal{H}(q_{\mathbf{\Lambda}})$, $\boldsymbol{h}_k(\mathbf{\Lambda}) := \nabla_{\boldsymbol{S}_k^{-1}}\mathcal{L}_0(\mathbf{\Lambda})$ and $\boldsymbol{\eta}_k(\mathbf{\Lambda}) := \nabla_{\boldsymbol{S}_k^{-1}}\mathcal{H}(q_{\mathbf{\Lambda}})$.

The entropy gradients can be expressed as expectations with respect to component distributions:

$$\begin{aligned}
\varphi_k(\mathbf{\Lambda}) &= -(\mathbb{E}_{q_{\boldsymbol{\lambda}_k}}[\log q_{\mathbf{\Lambda}}(\boldsymbol{X})] - \mathbb{E}_{q_{\boldsymbol{\lambda}_K}}[\log q_{\mathbf{\Lambda}}(\boldsymbol{X})]), \\
\boldsymbol{\gamma}_k(\mathbf{\Lambda}) &= -\pi_k \mathbb{E}_{q_{\boldsymbol{\lambda}_k}}[\nabla_{\boldsymbol{X}} \log q_{\mathbf{\Lambda}}(\boldsymbol{X})], \\
\boldsymbol{\eta}_k(\mathbf{\Lambda}) &= -\pi_k \mathbb{E}_{q_{\boldsymbol{\lambda}_k}}[\nabla^2_{\boldsymbol{X}} \log q_{\mathbf{\Lambda}}(\boldsymbol{X})],
\end{aligned}$$

These gradients can be estimated using Monte Carlo approximations $\widehat{\varphi}_k(\boldsymbol{X}_{1:B}^{(k)}; \mathbf{\Lambda})$, $\widehat{\boldsymbol{\gamma}}_k(\boldsymbol{X}_{1:B}^{(k)}; \mathbf{\Lambda})$ and $\widehat{\boldsymbol{\eta}}_k(\boldsymbol{X}_{1:B}^{(k)}; \mathbf{\Lambda})$ based on i.i.d. samples $\boldsymbol{X}_{1:B}^{(k)}$ from $q_{\boldsymbol{\lambda}_k}$.

For the gradients of $\mathcal{L}_0(\mathbf{\Lambda})$, an empirical approximation is used like in previous sections.

Specifically,

$$\widehat{f}_k(\boldsymbol{X}_{1:N};\boldsymbol{\Lambda}) = \frac{1}{N}\sum_{i=1}^{N}(q_{\boldsymbol{\lambda}_k}(\boldsymbol{X}_i) - q_{\boldsymbol{\lambda}_K}(\boldsymbol{X}_i)),$$

$$\widehat{\boldsymbol{g}}_k(\boldsymbol{X}_{1:N};\boldsymbol{\Lambda}) = \frac{\pi_k}{N}\sum_{i=1}^{N}\boldsymbol{S}_k(\boldsymbol{X}_i - \boldsymbol{\mu}_k)q_{\boldsymbol{\lambda}_k}(\boldsymbol{X}_i),$$

$$\widehat{\boldsymbol{H}}_k(\boldsymbol{X}_{1:N};\boldsymbol{\Lambda}) = \frac{\pi_k}{N}\sum_{i=1}^{N}(\boldsymbol{S}_k(\boldsymbol{X}_i - \boldsymbol{\mu}_k)(\boldsymbol{X}_i - \boldsymbol{\mu}_k)^T\boldsymbol{S}_k - \boldsymbol{S}_k)q_{\boldsymbol{\lambda}_k}(\boldsymbol{X}_i).$$

Algorithm. Algorithm 3 gives an implementation for Gaussian mixture NVA (eNVA-GM) when N is large and $\widehat{f}(\boldsymbol{X}_{1:N};\boldsymbol{\Lambda})$, $\widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:N};\boldsymbol{\Lambda})$, and $\widehat{\boldsymbol{H}}(\boldsymbol{X}_{1:N};\boldsymbol{\Lambda})$, are too costly to compute, instead using the Monte Carlo approximations $\widehat{f}(\boldsymbol{X}_{1:B}^{(b)};\boldsymbol{\Lambda})$, $\widehat{\boldsymbol{g}}(\boldsymbol{X}_{1:B}^{(b)};\boldsymbol{\Lambda})$ and $\widehat{\boldsymbol{H}}(\boldsymbol{X}_{1:B}^{(b)};\boldsymbol{\Lambda})$, where $\boldsymbol{X}_{1:B}^{(b)}$ is a sequence of size B consisting of vectors sampled i.i.d. from the elements of $\boldsymbol{X}_{1:n}$.

The hyperparameters are the number of iterations T, the number of samples B used to compute the gradients, the number of components K used in the Gaussian mixture, the initial parameters $\boldsymbol{\theta}_0 = (\pi_1, \ldots, \pi_{K-1}, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K, \boldsymbol{S}_1, \ldots, \boldsymbol{S}_K)$, the learning rate schedule $(\rho_t)_{t\in[T]}$, the learning rate schedule $(\omega_t)_{t\in[T]}$, and a lower bound for the covariance matrix eigenvalues α.

4.2 Interpretation of eNVA-GM as a multimodal optimization algorithm

Multiplicity of search distributions. In Gaussian NVA, the fitted Gaussian distribution behaves like a single particle moving on the landscape of p according to the natural gradient dynamics. Similarly, when using a mixture of Gaussian distributions, each component of the mixture can be interpreted as a particle exploring the landscape. This multiplicity allows to estimate multiple modes, with the K components acting as search agents. However, using Gaussian mixture NVA differs fundamentally from running multiple instances of Gaussian NVA in parallel. Unlike independent particles, the components of the mixtures are coupled through the entropy penalty, which governs their dynamics involving the whole mixture.

Role of the entropy penalty. The entropy term in the variational objective promotes dispersion of the mixture. This has a double effect: it induces a repulsive force between the components, encouraging them to separate and avoid collapsing into the same mode, and it forces individual components to spread by increasing the eigenvalues of their covariance matrices. The repulsive effect of the entropy penalty plays a crucial role in preventing multiple components from converging to the same mode. The intensity of this repulsive

Algorithm 3: Gaussian mixture eNVA (eNVA-GM)

Given samples $\boldsymbol{X}_{1:N}$.
Set T, B, K, $\boldsymbol{\theta}_0$, $(\rho_t)_{t\in[T]}$, $(\omega_t)_{t\in[T]}$, α.
Compute $(v_{k,0})_{k\in[K-1]} = (\log(\pi_{k,0}/\pi_{K,t}))_{k\in[K-1]}$.
for $t = 0\!:\!(T-1)$ **do**
  Sample $\boldsymbol{X}_i^{(b)} \overset{\text{i.i.d.}}{\sim} \mathcal{U}(\{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n\})$, for $i = 1\!:\!B$.
  for $k = 1\!:\!K$ **do**
    Compute $\widehat{\boldsymbol{\gamma}}_k(\boldsymbol{X}_{1:B}^{(k)}; \boldsymbol{\Lambda}_t)$, $\widehat{\boldsymbol{\eta}}_k(\boldsymbol{X}_{1:B}^{(k)}; \boldsymbol{\Lambda}_t)$.
    Sample $\boldsymbol{X}_i^{(k)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\boldsymbol{\mu}_{k,t}, \boldsymbol{S}_{k,t}^{-1})$, for $i = 1\!:\!B$.
    Compute $\widehat{\boldsymbol{g}}_k(\boldsymbol{X}_{1:B}^{(b)}; \boldsymbol{\Lambda}_t)$, $\widehat{\boldsymbol{H}}_k(\boldsymbol{X}_{1:B}^{(b)}; \boldsymbol{\Lambda}_t)$.
    Update
      $\boldsymbol{S}_{k,t+1} = \boldsymbol{S}_{k,t} - 2\rho_t \pi_{k,t}^{-1}(\widehat{\boldsymbol{H}}_k(\boldsymbol{X}_{1:B}^{(b)}; \boldsymbol{\Lambda}_t) + \omega_t \widehat{\boldsymbol{\eta}}_k(\boldsymbol{X}_{1:B}^{(k)}; \boldsymbol{\Lambda}_t))$,
      $\boldsymbol{S}_{k,t+1} = (\boldsymbol{S}_{k,t+1}^{-1} + \alpha \boldsymbol{I})^{-1}$,
      $\boldsymbol{\mu}_{k,t+1} = \boldsymbol{\mu}_{k,t} + \rho_t \pi_{k,t}^{-1} \boldsymbol{S}_{k,t+1}^{-1}(\widehat{\boldsymbol{g}}_k(\boldsymbol{X}_{1:B}^{(b)}; \boldsymbol{\lambda}_t) + \omega_t \widehat{\boldsymbol{\gamma}}_k(\boldsymbol{X}_{1:B}^{(k)}; \boldsymbol{\Lambda}_t))$.
  end
  for $k = 1\!:\!(K-1)$ **do**
    Compute $\widehat{f}_k(\boldsymbol{X}_{1:B}^{(b)}; \boldsymbol{\Lambda}_t)$, $\widehat{\varphi}_k(\boldsymbol{X}_{1:B}^{(k)}; \boldsymbol{\Lambda}_t)$.
    Update $v_{k,t+1} = v_{k,t} + \rho_t(\widehat{f}_k(\boldsymbol{X}_{1:B}^{(b)}; \boldsymbol{\Lambda}_t) + \omega_t \widehat{\varphi}_k(\boldsymbol{X}_{1:B}^{(k)}; \boldsymbol{\Lambda}_t))$.
  end
end
Compute $(\pi_{k,T})_{k\in[K]}$ from $(v_{k,T})_{k\in[K-1]}$.
return $\boldsymbol{\theta}_T$.

ω_∞	0	> 0	0
α	0	0	> 0
$\boldsymbol{S}_{k,\infty}^{-1}$	0	$\approx -\omega_\infty(\nabla^2 p(\boldsymbol{x}_i^*))^{-1}$	$\approx \alpha \boldsymbol{I}$
KDE error	high	optimizable	optimizable
Repulsion error	0	yes	0

Table 2: Errors on the estimation of a mode $\boldsymbol{x}_i^*$ of p, depending on the covariance matrix lower bound α and whether the annealing schedule goes to 0. We assume that the k-th component mean $\boldsymbol{\mu}_{k,t}$ of the search mixture converges to $\boldsymbol{x}_i^*$. The repulsion bias exists when ω_∞ but its magnitude is problem-dependent. The error induced by the kernel density estimate always exists, but it can be optimized by adjusting the limiting covariance matrix $\boldsymbol{S}_{k,\infty}^{-1}$ using ω_∞ or α.

effect is regulated by the temperature parameter, controlling the relative weight of the entropy penalty in the objective function.

Importance of annealing and damping. The entropy penalty introduces two sources of errors. The first one is similar to the KDE error in Gaussian NVA, concerning the covariance matrices. If the covariance matrices do not shrink to 0, as is the case when a positive temperature is maintained, then the optimal parameters $\boldsymbol{\Lambda}^{*,\omega}$ maximizing $\mathcal{L}_\omega(\boldsymbol{\Lambda})$, can yield component means ($\boldsymbol{\mu}^{*,\omega}$) that deviate from the true modes of p. This deviation becomes more pronounced as ω increases. The second source of error arises from the repulsive effect induced by the entropy penalty, which pushes the component means apart, introducing a bias by driving them away from the modes their target modes.

Table 2 summarizes the impact of the limit of the annealing schedule and the damping factor on the two types of errors. While the KDE error cannot be completely avoided, it can be minimized by tuning the limit of the annealing schedule ω_∞ and the damping factor α. In contrast, the repulsion bias due to the entropy penalty can only be completely eliminated when $\omega_\infty = 0$. Unlike in Gaussian eNVA, where the KDE bias-variance trade-off could be optimized independently using ω_∞ or α, in eNVA-GM, ω_∞ is required to cancel the repulsion bias, leaving α as the only parameter to mitigate the KDE error.

The repulsion bias can be acceptable if it is small relative to the KDE error. For instance, if the mixture components are sufficiently well-separated while maintaining positive covariance matrices, the entropy of the mixture is less sensitive to the distances between component means and much more dependent on their individual covariance matrices. In such cases, the repulsion force between component means can be negligible. However, this scenario is not guaranteed, justifying the necessity of a damping step to manage these two errors effectively.

5 Conclusion

This paper introduces empirical natural variational annealing (eNVA), as a new framework for empirical mode estimation, with applications in both unimodal and multimodal problems. By extending the natural variational annealing (NVA) framework, eNVA effectively provides a mechanism to balance exploration and exploitation through temperature annealing, while maintaining the computational efficiency of natural gradient-based updates. In unimodal cases, Gaussian eNVA generalizes the Gaussian mean-shift algorithm, incorporating more sophisticated search strategies.

The Gaussian mixture eNVA (eNVA-GM) is particularly suited for addressing multimodal problems. It leverages the diversity of mixture components promoted by an entropy penalty to track multiple modes in complex landscapes. The coupling between components through the entropy penalty distinguishes eNVA-GM from parallel mode estimation methods, ensuring better exploration of the search space.

Further work includes developing adaptive annealing schedules to optimize the search and the bias-variance trade-off induced by the kernel density estimation and the mixture entropy term. In addition, the flexibility of eNVA given by the choice of the variational family can be exploited to develop scalable versions of eNVA to investigate new applications in high-dimensional contexts.

References

Amari, S.-I. (1998). Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276.

Arias-Castro, E., Mason, D., and Pelletier, B. (2016). On the estimation of the gradient lines of a density and the consistency of the mean-shift algorithm. *Journal of Machine Learning Research*, 17:1–28.

Bonnabel, S. (2013). Stochastic gradient descent on Riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229.

Carreira-Perpiñan, M. Á. (2006). Acceleration strategies for Gaussian mean-shift image segmentation. *Conference on Computer Vision and Pattern Recognition*, 1:1160–1167.

Carreira-Perpiñan, M. Á. (2007). Gaussian mean-shift is an EM algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(5):767–776.

Chacón, J. E. (2015). A population background for nonparametric density-based clustering. *Statistical Science*, 30(4):518–532.

Chacón, J. E. and Monfort, P. (2014). A comparison of bandwidth selectors for mean shift clustering. In *Theoretical and Applied Issues in Statistics and Demography*. ISAST.

Chaudhuri, P. and Marron, J. S. (1999). Sizer for exploration of structures in curves. *Journal of the American Statistical Association*, 94(447):807–823.

Chen, X., Zhou, Y., Huang, X., and Li, C. (2008). Adaptive bandwidth mean shift object tracking. *Conference on Robotics, Automation and Mechatronics*.

Chernoff, H. (1964). Estimation of the mode. *Annals of the Institute of Statistical Mathematics*, 16(1):31–41.

Comaniciu, D. (2003). An algorithm for data-driven bandwidth selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(2):281–288.

Comaniciu, D. and Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619.

Dalenius, T. (1965). The mode–A neglected statistical parameter. *Journal of the Royal Statistical Society Series A: General*, 128(1):110–117.

Fukunaga, K. and Hostetler, L. (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 21(1):32–40.

Genovese, C. R., Perone-Pacifico, M., Verdinelli, I., and Wasserman, L. (2016). Nonparametric inference for density modes. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78(1):99–126.

Honkela, A., Tornio, M., Raiko, T., and Karhunen, J. (2007). Natural conjugate gradient in variational inference. *International Conference on Neural Information Processing*.

Kronmal, R. and Tarter, M. (1968). The estimation of probability densities and cumulatives by Fourier series methods. *Journal of the American Statistical Association*, 63(323):925–952.

Le Minh, T., Arbel, J., Möllenhoff, T., Khan, M. E., and Forbes, F. (2025). Natural variational annealing for multimodal optimization. *arXiv preprint arXiv:2501.04667*.

Lin, W., Khan, M. E., and Schmidt, M. (2019). Fast and simple natural-gradient variational inference with mixture of exponential-family approximations. *International Conference on Machine Learning*.

Liu, B. and Ghosh, S. K. (2020). On empirical estimation of mode based on weakly dependent samples. *Computational Statistics & Data Analysis*, 152:107046.

Loftsgaarden, D. O. and Quesenberry, C. P. (1965). A nonparametric estimate of a multivariate density function. *The Annals of Mathematical Statistics*, 36(3):1049–1051.

Minnotte, M. C. and Scott, D. W. (1993). The mode tree: A tool for visualization of nonparametric density features. *Journal of Computational and Graphical Statistics*, 2(1):51–68.

Parzen, E. (1962). On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076.

Sager, T. W. (1978). Estimation of a multivariate mode. *The Annals of Statistics*, 6(4):802–812.

Samworth, R. J. (2018). Recent progress in log-concave density estimation. *Statistical Science*, 33(4):493–509.

Sato, M.-A. (2001). Online model selection based on the variational Bayes. *Neural Computation*, 13(7):1649–1681.

Venter, J. H. (1967). On estimation of the mode. *The Annals of Mathematical Statistics*, 38(5):1446–1455.

Zhao, Q., Yang, Z., Tao, H., and Liu, W. (2009). Evolving mean shift with adaptive bandwidth: A fast and noise robust approach. *Asian Conference on Computer Vision*.